ReadMe

Repository name - vikascsepandey/vikas

This docker repository contains image containing all the codes and requirement environment tag with the image.

Step1 -
Pull Repository - sudo docker pull vikascsepandey/vikas
After pull you can easily run the image file by typing following command

Step2 -
Run - sudo docker run -it vikascsepandey/vikas

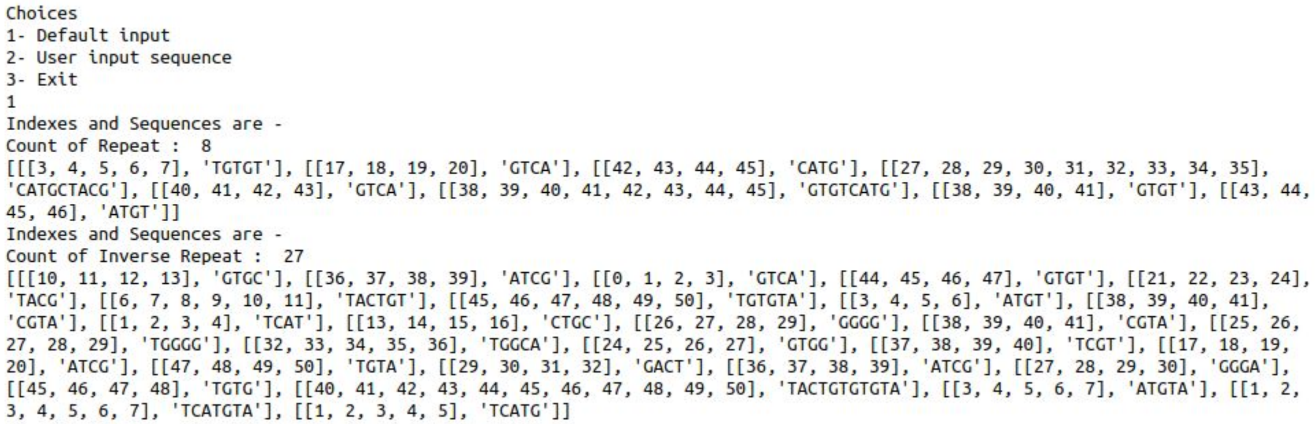
Docker image contains - An OS ubuntu, Vim editor, Python with version 3, Required python libraries namely (sklearn, pandas, numpy, matplotlib), gpsr docker required files for question 3, which is bundle as a image.

Step3 -
After Run image do **ls** for list the directories inside it.

Step4 -
Move inside the **Assign** folder which contains all the required codes in folder Q1, Q2, Q3 and Q4.

Run Question 1-

Q1 contains the required python file to run this run command
python 3 A2_Q1.py
After run the command the following option comes up you have to choose one.
If you want to run the code for **given sequence** in question you can choose **1**. For a new sequence you can choose **2** and give a input sequence and if you want to quit the program press **3**.



```
Choices
1- Default input
2- User input sequence
3- Exit
1
Indexes and Sequences are -
Count of Repeat :  8
[[[3, 4, 5, 6, 7], 'TGTGT'], [[17, 18, 19, 20], 'GTCA'], [[42, 43, 44, 45], 'CATG'], [[27, 28, 29, 30, 31, 32, 33, 34, 35],
'CATGCTACG'], [[40, 41, 42, 43], 'GTCA'], [[38, 39, 40, 41, 42, 43, 44, 45], 'GTGTCATG'], [[38, 39, 40, 41], 'GTGT'], [[43, 44,
45, 46], 'ATGT']]
Indexes and Sequences are -
Count of Inverse Repeat :  27
[[[10, 11, 12, 13], 'GTGC'], [[36, 37, 38, 39], 'ATCG'], [[0, 1, 2, 3], 'GTCA'], [[44, 45, 46, 47], 'GTGT'], [[21, 22, 23, 24],
'TACG'], [[6, 7, 8, 9, 10, 11], 'TACTGT'], [[45, 46, 47, 48, 49, 50], 'TGTGTA'], [[3, 4, 5, 6], 'ATGT'], [[38, 39, 40, 41],
'CGTA'], [[1, 2, 3, 4], 'TCAT'], [[13, 14, 15, 16], 'CTGC'], [[26, 27, 28, 29], 'GGGG'], [[38, 39, 40, 41], 'CGTA'], [[25, 26,
27, 28, 29], 'TGGGG'], [[32, 33, 34, 35, 36], 'TGGCA'], [[24, 25, 26, 27], 'GTGG'], [[37, 38, 39, 40], 'TCGT'], [[17, 18, 19,
20], 'ATCG'], [[47, 48, 49, 50], 'TGTA'], [[29, 30, 31, 32], 'GACT'], [[36, 37, 38, 39], 'ATCG'], [[27, 28, 29, 30], 'GGGA'],
[[45, 46, 47, 48], 'TGTG'], [[40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50], 'TACTGTGTGTA'], [[3, 4, 5, 6, 7], 'ATGTA'], [[1, 2,
3, 4, 5, 6, 7], 'TCATGTA'], [[1, 2, 3, 4, 5], 'TCATG']]
```

In the output here first i give the count corresponding to Repeat sequence and in the list of list structure first list give the index of sequence **length >=4** and second list the corresponding sequence. Similarly for Inverse repeat.

Run Question 2-

To run the question 2 the needed files is directly pass by the program which is stored in Q2/A2_q2_Generated_Train_data for amino acid composition and Q2/final_amino_acid_result.csv for atomic composition.

To run the code type -

Go to the folder Q3 and type -

python3 A2_Q2.py

It take little time to plot the cluster from both the dataset for k-means and agglomerative clustering and result comes into the form of cluster in .png file in same Q2 folder.

Run Question 3-

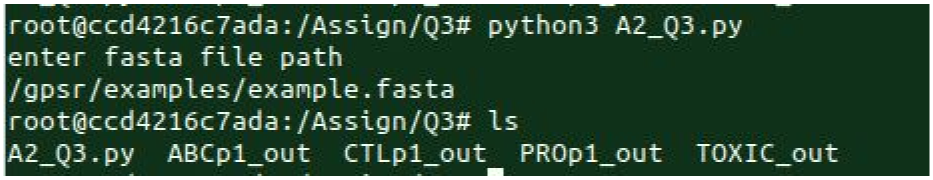
To run the question 3

Go to the folder Q3 and type -

python3 A2_Q3.py

Give the fasta file path which will be the input

Output files will be generated in q3 folder



```
root@ccd4216c7ada:/Assign/Q3# python3 A2_Q3.py
enter fasta file path
/gpsr/examples/example.fasta
root@ccd4216c7ada:/Assign/Q3# ls
A2_Q3.py  ABCp1_out  CTLp1_out  PROp1_out  TOXIC_out
```

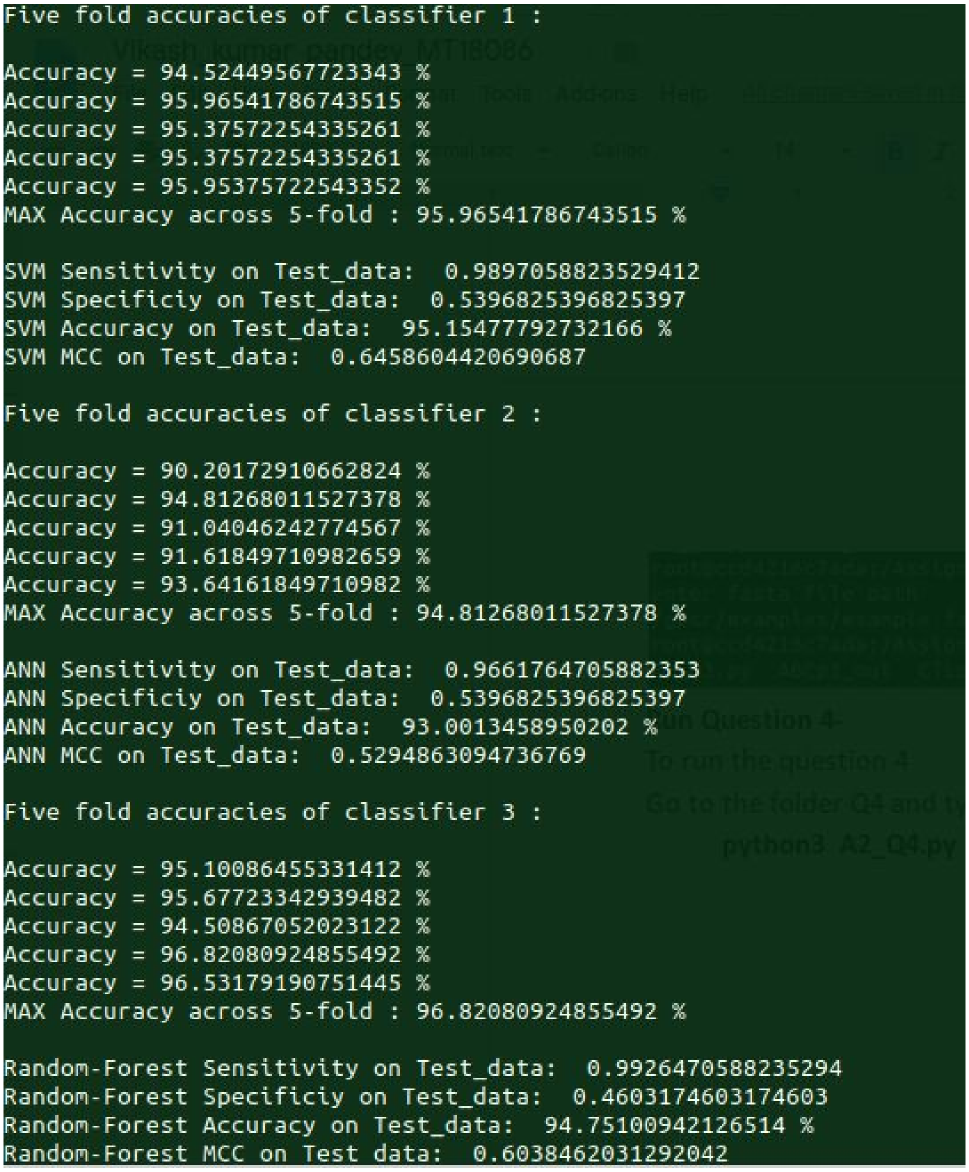
Run Question 4-

To run the question 4

Go to the folder Q4 and type -

python3 A2_Q4.py

Output -



```
Five fold accuracies of classifier 1 :

Accuracy = 94.52449567723343 %
Accuracy = 95.96541786743515 %
Accuracy = 95.37572254335261 %
Accuracy = 95.37572254335261 %
Accuracy = 95.95375722543352 %
MAX Accuracy across 5-fold : 95.96541786743515 %

SVM Sensitivity on Test_data:  0.9897058823529412
SVM Specificiy on Test_data:  0.5396825396825397
SVM Accuracy on Test_data:  95.15477792732166 %
SVM MCC on Test_data:  0.6458604420690687

Five fold accuracies of classifier 2 :

Accuracy = 90.20172910662824 %
Accuracy = 94.81268011527378 %
Accuracy = 91.04046242774567 %
Accuracy = 91.61849710982659 %
Accuracy = 93.64161849710982 %
MAX Accuracy across 5-fold : 94.81268011527378 %

ANN Sensitivity on Test_data:  0.9661764705882353
ANN Specificiy on Test_data:  0.5396825396825397
ANN Accuracy on Test_data:  93.0013458950202 %
ANN MCC on Test_data:  0.5294863094736769

Five fold accuracies of classifier 3 :

Accuracy = 95.10086455331412 %
Accuracy = 95.67723342939482 %
Accuracy = 94.50867052023122 %
Accuracy = 96.82080924855492 %
Accuracy = 96.53179190751445 %
MAX Accuracy across 5-fold : 96.82080924855492 %

Random-Forest Sensitivity on Test_data:  0.9926470588235294
Random-Forest Specificiy on Test_data:  0.4603174603174603
Random-Forest Accuracy on Test_data:  94.75100942126514 %
Random-Forest MCC on Test data:  0.6038462031292042
```